ULTERA.008A PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicants	:	Mohamad Nourmohamadian et al.	Group Art Unit 2185
Appl. No.	:	10/655,948	
Filed	:	09/05/2003	
For	:	VIRTUAL TAPE STACKER	
Examiner	:	Campos, Yaima	

APPEAL BRIEF

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22213-1450

Dear Sir:

Applicants in the above-captioned patent application are appealing the final rejection of all claims pending in a final Office action dated 01/31/2007. Claims 16-24 are pending and rejected. Claims 1-15 were previously canceled. An examiner panel decision dated 9/13/2007 affirmed the rejection of the pending claims after a pre-appeal brief review. The time to respond has been extended three months to 01/14/2008.

TABLE OF CONTENTS

REAL PARTY IN INTEREST

The real party in interest is the assignee Ultera Systems, Inc., a California corporation, 26052 Merit Circle, #106, Laguna Hills, CA 92653, hereinafter "Appellant."

RELATED APPEALS & INTERFERENCES

None of the Appellant or Appellant's legal representative is aware of any appeal or interference which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

STATUS OF CLAIMS

Claims 1-15 have been canceled. Claims 16-24 are currently pending and stand rejected. Claims 16-24 are the claims appealed herein and are set forth in the Claims Appendix attached hereto.

STATUS OF AMENDMENTS

Appellant filed an amendment subsequent to the final rejection of the claims, which was entered via an advisory action dated 04/16/2007. Appellant filed a Pre-Appeal Brief on 06/30/2007 corresponding to a request for review. Arguments presented in that brief were rejected by the examiner panel mailed 09/13/2007.

SUMMARY OF CLAIMED SUBJECT MATTER

Regarding independent claim 16, an aspect of a virtual tape stacker is a server interface 410, a random access storage device 330, a data path 450 and a controller 400. (Page 5, paragraph [0023], lines 3-7; paragraph [0024], lines 1-9; Figure 4). The server interface 410 is adapted to communicate with a server 110. (Paragraph [0024], lines 4-5; Figure 4). The data path 450 is adapted to communicate with the random access data storage device 330. The controller 400 is configured to transfer data between the server interface 410 and the random access data storage device 330 via the data path 450. (Paragraph [0024], lines 9-10; Figure 4). The controller 400, 470 manages the data on the random access data storage device 330 as a plurality of virtual tape volumes 500. (Paragraph [0024], lines 11-15; Figures 4, 5A-C). The controller 400, 470 also defines a virtual tape drive 610 for transferring data between the server and the virtual tape volumes. (Page 7, paragraph [0028], lines 5-9; page 9, paragraph [0034], lines 1-9; page 10, paragraph [0037], lines 1-6; Figures 6, 8B). The controller 400 further defines a sequential order 820 for loading the virtual tape volumes 500 into the virtual tape drive. (Page 9, paragraph [0034], lines 1-2; Figures 8A-B). In response to an eject command from the server 110, the controller 400 unloads one of the virtual tape volumes 500 from the virtual tape drive and loads a next consecutive one of the virtual tape volumes into the virtual tape drive according to the sequential order 820. (Page 9, paragraph [0034], lines 4-9; Figures 8A-B).

Regarding independent claim 19, another aspect of a virtual tape stacker is a plurality of virtual tape volumes 500 provided on a random access storage device 330. (Page 6, paragraph [0025], lines 1-2; Figures 4, 5A-C). A virtual tape drive 610 is defined in a volume management table 601 located on the random access storage device 330, 501. (Page 7, paragraph [0028], lines 5-6; Figures 5A-C, 6). The virtual tape volumes 640 are identified in a plurality of data management tables 602 located on the random access storage device. (Paragraph [0029], lines 1-2; Figures 5B, 6). The volume management table 601 stores a plurality of pointers 620 to the data management tables 602 so as to identify the location of the virtual tape volumes 640. (Paragraph [0028], lines 5-6, 9-10; paragraph [0029], lines 1-5; Figure 6). An access order for the pointers 620 is predetermined so as to define a sequential order 820 for

loading the virtual tape volumes 500 into the virtual tape drive in response to eject commands from a server. (Page 9, paragraph [0034], lines 1-10; pages 23-24, paragraph [0070], lines 11-25; Figures 6, 8A-B, 23).

Regarding independent claim 22, a further aspect of a virtual tape stacker comprises a plurality of virtual tape volumes 500, a virtual tape drive 610 and a virtual tape manager 470. (Figures 4, 5A-C, 6). The virtual tape volumes 500 are configured on a random access data storage device 330. (Page 6, paragraph [0025], lines 1-6). The virtual tape drive 610 is defined by a controller 400 in communications with the random access data storage device 330. (Page 5, paragraph [0024], lines 1-4, 11-15; page 7, paragraph [0028], lines 1-9; Figures 4, 6). The virtual tape manager 470 is configured on the controller 400 so as to transfer data between one of the virtual tape volumes 500 loaded into the virtual tape drive and an application program. (Page 6, paragraph [0026], lines 1-3, 10-17; Figure 5B). The virtual tape manager 470 indicates a sequential order 820 for loading a next consecutive one of the virtual tape volumes 500 into the virtual tape drive upon ejection of the loaded one of the virtual tape volumes. (Page 9, paragraph [0034], lines 1-9; Figures 8A-B).

GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

Claims 16-24 were rejected under 35 U.S.C. §102(b) as being anticipated by Keele (US Patent 5,455,926). Claim 24 was further rejected under 35 U.S.C. §103(a) as unpatentable over Keele in view of Dailey (US 2004/0098244).

ARGUMENT

Conventional sequential stackers are well-known for automatic data back-up and retrieval applications. A conventional sequential stacker has a tape drive, tape cartridge slots and a robotic mechanism that moves tapes between the slots and the tape drive. Unlike a tape library, a sequential stacker's robotic mechanism autoloads tape cartridges between the tape drive and the tape cartridge slots in a predetermined sequential order.

Appellant claims a virtual tape stacker with virtual tape volumes organized in sequential order, where a next sequential virtual tape volume is automatically loaded when the current virtual tape volume is ejected. Unlike Appellant's claimed virtual tape stacker, Keele describes an optical disk library that emulates a tape library; and like a tape library, Keele does not teach, disclose or suggest, either explicitly or inherently, a mechanism that autoloads virtual tapes into a virtual tape drive in a predetermined sequential order. Accordingly, the examiner does not meet his *prima facie* burden as set forth under separate headings numbered 1 and 2, below.

1. Keele Does Not Teach All of the Claim Limitations

Each of claims 16-24 are separately identified and argued below and do not stand or fall together.

Claim 16

Claim 16 cites, in part: "the <u>controller defines a sequential order for loading</u> the virtual tape volumes into the virtual tape drive . . . <u>in response to an eject command</u> from the server, the controller unloads one of the virtual tape volumes from the virtual tape drive and <u>loads a next consecutive one of the virtual tape volumes into the virtual tape drive according to the sequential order</u>." (Emphasis added.)

The functional portions of claim 16 cannot be ignored. *ACCO Brands, Inc. v. Micro Security Devices, Inc.*, 346 F.3d 1075, 1078, 68 USPQ2d 1526, 1528 (Fed. Cir. 2003) ("The functional language is, of course, an additional limitation in the claim.").

The examiner goes through a lengthy argument with respect to each independent claim (16, 19 and 22) to assert that Keele discloses loading virtual tapes in

a sequential order. See, for example: "Keele's invention is able to load/unload a virtual tape that is next in the sequential order in which tapes are written to optical disks." Final Office action, page 8, lines 8-9. See also page 5, lines 1-2; page 12, lines 1-2. The examiner does not establish that Keele teaches, explicitly or inherently, a controller that defines a sequential order for loading virtual tape volumes. The examiner merely asserts that Keele is able to load/unload a virtual tape in the order in which they are written to optical disks. Because Keele optical disks are random access devices, they are able to load/unload virtual tape volumes in any order. But Keele does not teach that the controller loads/unloads volumes in any specific order, much less a defined sequential order.

Further, the examiner does not establish that Keele teaches, explicitly or inherently, a controller that unloads a virtual tape volume and loads a next consecutive one in response to an eject command. Indeed, Keele requires the opposite, i.e. virtual tapes are loaded according to the receipt of a VSN (volume and serial number): "When MOST receives a mount message, it automatically locates the requested VSN and mounts it in the drive." Keele, column 37, lines 37-39, emphasis added. Nothing in Keele describes that these VSN requests are made in any particular order or that a VSN is requested for a next consecutive virtual tape in response to an eject command. As such, Keele does not read on claim 16 .

Claim 17

Claim 17 cites, in part: "a volume management table residing on the random access data storage device and accessible by the controller, the volume management table having pointers to the virtual tape volumes; and a virtual tape manager residing on the controller that accesses the pointers so as to determine the next consecutive one of the virtual tape volumes.

Keele does not read on claim 17. The examiner equates the Keele tape map pointers in the Keele tape directory to the claimed pointers to virtual tape volumes. Final Office action, page 5, lines 9-12. In Keele, however, the tape map pointers are not used to determine the next consecutive virtual tape volume. Rather, Keele teaches that software or an operator determine which virtual tape to load through the VSN

(volume and serial number). Only after a virtual tape is loaded do the tape map pointers come into play so as to load the tape map into the controller. "The tape map pointer 334 [Fig. 5] points to a respective tape map 348 of each virtual tape. . . . When a virtual tape VSN is mounted a copy of the tape map 340 is read into the controller 14."

Keele, column 44, lines 5-17, emphasis added.

Nothing in Keele describes that the VSN requests are determined by accessing the tape map pointers. The opposite is the case, i.e. when the virtual tape having the requested VSN is mounted, then the tape map identified by the tape map pointer is read into the controller. Accordingly, the examiner does not establish that Keele reads on claim 17.

Claim 18

Claim 18 cites, in part: "a physical tape device; and a tape cartridge loadable into the physical tape device, wherein a physical tape volume corresponding to the tape cartridge is integrated into the virtual tape volume storage rotation." For example, the Appellant's specification describes the claim elements underlined above as: "[0035] As shown in FIG. 8A, one or more physical tape drives 350 may be incorporated into the virtual sequential stacker 800. . . The physical tape volume 840 automatically becomes part of the virtual tape volume storage rotation. After the last virtual tape volume 814 is un-mounted, the next tape to load into the virtual tape drive will be a write protected physical tape volume 840. . . . When the last physical tape volume 840 is un-mounted, the first virtual tape volume 812 is automatically loaded into the virtual tape drive."

In rejecting claim 18, the examiner only asserts that the Keele MOST system can be used with a physical tape device; Final Office action, page 6, lines 6-8; and that data can be transferred between physical tape and optical disk; lines 15-18. The examiner does not establish that the Keele physical tape volume becomes part of the virtual tape volume storage rotation as claimed. As such, the examiner does not establish that Keele reads on claim 18.

Claim 19

Claim 19 cites, in part: "storing in the volume management table a plurality of pointers to the data management tables so as to identify the location of the virtual tape volumes; and predetermining an access order for the pointers so as to define a sequential order for loading the virtual tape volumes into the virtual tape drive in response to eject commands from a server."

With respect to the claimed sequential order for loading virtual tape volumes into the virtual tape drive in response to eject commands, the examiner makes the same argument cited above with respect to claim 16. Final Office action, page 8, lines 8-9. That argument does not meet the examiner's prima facie burden of anticipation with respect to claim 19 as similarly argued with respect to claim 16, above.

With respect to the claimed predetermining step, the examiner equates the Keele tape map pointers in the Keele tape directory to the claimed pointers to data management tables. Final Office action, page 7, lines 8-11. In Keele, however, there is not a predetermined access order for the tape map pointers that define the sequential order for loading virtual tape volumes into the virtual tape drive. The examiner does not even assert that the Keele tape map pointers have a predetermined access order or that a tape map pointer access order defines a sequential order for loading virtual tape volumes. As such, the examiner does not establish that Keele reads on claim 19.

Claim 20

Claim 20 cites, in part: "reading one of the pointers according to the access order; locating one of the data management tables according to the read pointer; and addressing a next consecutive one in the sequential order of the virtual tape volumes according to the located one of the data management tables." The examiner does not cite any portion of Keele that teaches tape map pointers that have a predetermined access order, as noted with respect to claim 19, above. As such, Keele does not disclose that the tape map pointers are read according to that access order, as claimed. Hence the examiner does not establish that Keele reads on claim 20.

Claim 21

Claim 21 cites, in part: "providing a physical tape volume loaded on a physical tape device; and integrating the physical tape volume in a storage rotation of the virtual tape volumes." As argued with respect to claim 18, above, the examiner does not establish that a Keele physical tape volume becomes part of a virtual tape volume storage rotation. The physical tape volume is likewise not integrated in the storage rotation of the virtual tape volumes, as claimed here. Hence, the examiner does not establish that Keele reads on claim 21.

Claim 22

Claim 22 cites, in part: "wherein the virtual tape manager indicates a sequential order for loading a next consecutive one of the virtual tape volumes into the virtual tape drive upon ejection of the loaded one of the virtual tape volumes." With respect to these claim elements, the examiner makes the same argument cited above for claim 16. Final Office action, page 12, lines 1-2. That argument does not meet the examiner's prima facie burden of anticipation with respect to claim 22 as similarly argued with respect to claim 16, above.

Claim 23

Claim 23 cites, in part: "a volume management table maintained in the virtual tape manager, a plurality of pointers to the virtual tape volumes stored in the volume management table, wherein the sequential order of loading the virtual tape volumes into the virtual tape drive is determined by an access order of the pointers." The examiner equates the Keele tape map pointers in the Keele tape directory to the claimed pointers to virtual tape volumes. Final Office action, page 12, lines 12-15. As argued with respect to claim 17, above, the tape map pointers of Keele are not used to determine the next consecutive virtual tape volume. Rather, Keele teaches that software or an operator determine which virtual tape to load through the VSN. Only after a virtual tape is loaded do the tape map pointers come into play so as to load the tape map into the controller.

Further, Keele does not disclose an access order for the tape map pointers. The examiner provides a one page argument with respect to this element without citing any portion of Keele that even suggests that the Keele tape map pointers have an access order or that a sequential order of loading virtual tape volumes is determined by that access order as claimed. See final Office action, page 13. Therefore, the examiner does not establish that Keele reads on claim 23.

Claim 24

Claim 24 cites, in part: "a physical tape volume, wherein a last one of the virtual tape volumes is previous to the physical tape volume in the sequential access order and a first one of the virtual tape volumes is next from the physical tape volume in the sequential access order." As argued with respect to claim 18, above, the examiner does not establish with respect to Keele that a physical tape volume becomes part of the virtual tape volume storage rotation. Likewise here, the examiner specifically does not establish that the last virtual tape volume is previous to a physical tape volume in an access order and a first virtual tape volume is next from the physical tape volume in access order, as claimed. Indeed, in the examiners specific argument regarding this element, which is set forth without any citation to Keele, the examiner fails to identify with respect to Keele, a corresponding first and last virtual tape volume with respect to a sequential access order, as claimed. See final Office action, page 14, lines 18-22; page 15, lines 1-4. As such, the examiner does not establish that Keele reads on claim 24.

2. There Is No Properly Asserted 35 USC §103(a) Rejection

The examiner asserts that Keele anticipates claim 24. In asserting that claim 24 is unpatentable over Keele in view of Dailey, the examiner repeats the anticipation argument verbatim. *See* final Office action, pages 14-15; *cf.* pages 16-17. Following the repeated anticipation text, the examiner states the apparent purpose for citing Dailey (emphasis added): "To further detail Keele's disclosure, Dailey discloses incorporating physical tape volumes within virtual tape volumes."

To establish a *prima facie* case of obviousness, the prior art reference (or references when combined) must teach or suggest all the claim limitations. MPEP §2142. Here, the examiner does not assert that Dailey teaches or suggests a specific element of claim 24. As such, and in view of the simultaneous assertion that all claim 24 limitations are disclosed in Keele, the examiner has not stated a prima facie case of obviousness. Nor has the examiner given Appellant an opportunity to argue against this rejection. Specifically, Appellant can only dispute that Dailey does not "further detail Keele's disclosure."

Such boilerplate rejections fall far short of the examiner's burden of proof. *In re Oetiker,* 977 F.2d 1443, 1449, 24 USPQ2d 1443, 1447 (Fed. Cir. 1992) ("The 'prima facie' case notion . . . was intended to leave no doubt among examiners that they must state clearly and specifically any objections (the prima facie case) to patentability, and give the applicant fair opportunity to meet those objections with evidence and argument."), emphasis added. Clearly the Appellant had no fair opportunity during prosecution to meet these objections with respect to claim 24.

CONCLUSIONS

For the reasons set forth above, the final rejections of all of appealed Claims 16-24 are improper and Appellant respectfully requests that each of these rejections be reversed.

Respectfully submitted,

LAW OFFICE OF GLENN R. SMITH

Dated: 01/13/2008 By: Glenn R Smith

Glenn R. Smith, Esq.
Registration No. 38,308
Attorney of Record
28626 Brookhill Road
Trabuco Canyon, CA 92679-1163
Telephone/Fax: (949) 709-7164

CLAIMS APPENDIX

CLAIMS ON APPEAL:

16. A virtual tape stacker comprising:

a server interface adapted to communicate with a server;

a random access data storage device;

a data path adapted to communicate with the random access data storage device; and

a controller configured to transfer data between the server interface and the random access data storage device via the data path;

wherein the controller manages the data on the random access data storage device as a plurality of virtual tape volumes,

wherein the controller defines a virtual tape drive for transferring data between the server and the virtual tape volumes,

wherein the controller defines a sequential order for loading the virtual tape volumes into the virtual tape drive, and

wherein, in response to an eject command from the server, the controller unloads one of the virtual tape volumes from the virtual tape drive and loads a next consecutive one of the virtual tape volumes into the virtual tape drive according to the sequential order.

17. The virtual tape stacker according to claim 16 further comprising:

a volume management table residing on the random access data storage device and accessible by the controller, the volume management table having pointers to the virtual tape volumes; and

a virtual tape manager residing on the controller that accesses the pointers so as to determine the next consecutive one of the virtual tape volumes.

18. The virtual tape stacker according to claim 17 further comprising:

a physical tape device; and

a tape cartridge loadable into the physical tape device,

wherein a physical tape volume corresponding to the tape cartridge is integrated into the virtual tape volume storage rotation.

19. A virtual tape stacker method comprising:

providing a plurality of virtual tape volumes on a random access storage device;

defining a virtual tape drive in a volume management table located on the random access storage device;

identifying the virtual tape volumes in a plurality of data management tables located on the random access storage device;

storing in the volume management table a plurality of pointers to the data management tables so as to identify the location of the virtual tape volumes; and

predetermining an access order for the pointers so as to define a sequential order for loading the virtual tape volumes into the virtual tape drive in response to eject commands from a server.

20. The virtual tape stacker method according to claim 19 further comprising:

reading one of the pointers according to the access order;

locating one of the data management tables according to the read pointer; and

addressing a next consecutive one in the sequential order of the virtual tape volumes according to the located one of the data management tables.

21. The virtual tape stacker method according to claim 20 further comprising:

providing a physical tape volume loaded on a physical tape device; and

integrating the physical tape volume in a storage rotation of the virtual tape volumes.

22. A virtual tape stacker comprising:

a plurality of virtual tape volumes configured on a random access data storage device;

a virtual tape drive defined by a controller in communications with the random access data storage device;

a virtual tape manager configured on the controller so as to transfer data between one of the virtual tape volumes loaded into the virtual tape drive and an application program,

wherein the virtual tape manager indicates a sequential order for loading a next consecutive one of the virtual tape volumes into the virtual tape drive upon ejection of the loaded one of the virtual tape volumes.

23. The virtual tape stacker according to claim 22 further comprising:

a volume management table maintained in the virtual tape manager,

a plurality of pointers to the virtual tape volumes stored in the volume management table,

wherein the sequential order of loading the virtual tape volumes into the virtual tape drive is determined by an access order of the pointers.

24. The virtual tape stacker according to claim 23 further comprising:

a physical tape volume,

wherein a last one of the virtual tape volumes is previous to the physical tape volume in the sequential access order and a first one of the virtual tape volumes is next from the physical tape volume in the sequential access order.

EVIDENCE APPENDIX

none

RELATED PROCEEDINGS APPENDIX

none